Notes on Adaptive Context Trees and Local Predictors for Atlas Neural Networks

Core Architecture

An Atlas neural network can be viewed as two largely independent components:

- **Context generator**
- Produces a binary or multi-way context from the input.
- Partitions the input space.
- Selects an appropriate model.
- **Local predictor**
- The composite linear mapping associated with a particular context.
- Models the function only within its assigned region.
- Makes the final prediction.

Conceptually,

```
Input
   │
   ▼
Context Generator
   │
Context ID
   │
   ▼
Local Predictor
   │
Prediction
```

This separation makes the architecture easier to reason about. The context generator determines *where* the input lies, while the local predictor determines *what function* should be applied there.

Local Predictors

The term **local predictor** is preferable to "linear layer" or "parameter set."

Advantages:

- Emphasizes that each predictor only approximates a local region.
- Does not constrain the predictor implementation.
- Future implementations could use:

- composite linear maps,
- small MLPs,
- low-rank models,
- polynomial approximations,
- lookup tables,
- or recursive Atlas modules.

Adaptive Context Depth

Conventional Atlas implementations may consume a fixed number of context bits.

Instead, context generation can terminate early.

```
Context bit 1
        │
        ▼
Region sufficiently homogeneous?
        │
   Yes ─┴──► Select predictor
        │
       No
        ▼
Context bit 2
```

Only difficult regions require additional context.

Potential advantages:

- lower average inference cost,
- fewer stored contexts,
- better parameter efficiency,
- shallower decisions in simple regions,
- deeper specialization only where required.

"If–Except–If" Interpretation

Context trees can be viewed as chains of increasingly specific exceptions.

Rather than repeatedly partitioning space, begin with a general rule and refine only when necessary.

Example:

```
If A:
    predict X
```

```
Except if B:
    predict Y

Except if C:
    predict Z
```

Each new proposition handles an exceptional subset of the previous rule.

Training therefore becomes constructive:

1. Start with a simple predictor.
2. Detect prediction failures.
3. Find an additional proposition that isolates the failures.
4. Create a more specialized local predictor.
5. Repeat only where necessary.

This avoids increasing model complexity in regions that are already well modelled.

Relationship to Classical Context Trees

Classical context trees partition **history**.

Atlas partitions **feature space**.

Classical:

```
history
   │
context tree
   │
prediction
```

Atlas:

```
feature vector
      │
context generator
      │
local predictor
      │
prediction
```

The underlying principle is identical:

> Progressively refine context until a sufficiently accurate local model exists.

The primary difference lies in how context is generated.

Classical context trees derive context from symbol history.

Atlas derives context from geometric tests such as random projections, binary threshold functions, or atan2 wedge indices.

Information-Theoretic View

Not all context variables contribute equal information.

If context variables are ordered by expected information gain or entropy, many samples should terminate after only a small number of tests.

This naturally combines:

- adaptive computation,
- adaptive memory usage,
- adaptive model complexity.

The resulting architecture resembles a variable-depth context tree over feature space.

Training Philosophy

The architecture should grow only where prediction error justifies additional complexity.

Rather than globally increasing network capacity,

- preserve successful local predictors,
- split only failing contexts,
- allocate new predictors only for statistically distinct exceptions.

Complexity therefore follows the observed structure of the data rather than being fixed a priori.

Geometric Interpretation

The term **Atlas** becomes particularly appropriate.

An atlas in differential geometry consists of many local coordinate charts that together describe a complicated manifold.

Similarly, an Atlas neural network consists of many local predictors whose domains are determined by the context generator.

The global function is represented not by a single universal model, but by a collection of specialized local models together with an adaptive routing mechanism.

This viewpoint cleanly separates *partitioning* from *prediction*, and suggests that future research should focus equally on improving context generation and improving the expressiveness of local predictors.